



THE UNITED STATES PATENT AND TRADEMARK OFFICE

1/15/08 DAC

In re Patent Application of

OKADA et al.

Atty. Ref.: MJS-723-951

Serial No. 09/722,410

TC/A.U.: 3714

Filed: November 28, 2000

Examiner: MOSSER, Robert

For: STORAGE DEVICES FOR VIDEO GAME SYSTEMS AND  
VIDEO GAME SYSTEM EMULATORS

\* \* \* \* \*

March 5, 2008

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Sir:

**RESPONSE TO DECISION ON PETITION UNDER 37 CFR 1.59**

Applicant is in receipt of the Decision on Petition mailed February 29, 2008 in the subject patent application. According to the Decision, the decision on Applicant's petition to expunge "will be held in abeyance until allowance of the application or mailing of an *Ex parte Quayle* action or a Notice of Abandonment, at which time the petition will be decided" (underlining in original omitted).

This application has now been allowed (copy of Notice of Allowance and Fee(s) Due attached) and thus Applicant respectfully requests a decision on the petition to expunge dated January 16, 2004.

No fee is believed to be associated with the filing of this paper. Nonetheless, should the USPTO determine that a fee is required, please charge such fee to our Deposit Account No. 14-1140.

Respectfully submitted,

**NIXON & VANDERHYE P.C.**

By:



Michael J. Shea  
Reg. No. 34,725

MJS:dbp  
901 North Glebe Road, 11th Floor  
Arlington, VA 22203-1808  
Telephone: (703) 816-4000  
Facsimile: (703) 816-4100



# UNITED STATES PATENT AND TRADEMARK OFFICE



UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

## NOTICE OF ALLOWANCE AND FEE(S) DUE

27562 7590 02/28/2008

NIXON & VANDERHYE, P.C.  
901 NORTH GLEBE ROAD, 11TH FLOOR  
ARLINGTON, VA 22203

EXAMINER

MOSSER, ROBERT E

ART UNIT

PAPER NUMBER

3714

DATE MAILED: 02/28/2008

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/722,410	11/28/2000	Satoru Okada	723-951	4624

TITLE OF INVENTION: MEMORY FOR VIDEO GAME SYSTEM AND EMULATOR USING THE MEMORY

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	\$1440	\$0	\$0	\$1440	05/28/2008

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

### HOW TO REPLY TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above.

If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is the same, pay the TOTAL FEE(S) DUE shown above.

B. If the status above is to be removed, check box 5b on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and twice the amount of the ISSUE FEE shown above, or

If the SMALL ENTITY is shown as NO:

A. Pay TOTAL FEE(S) DUE shown above, or

B. If applicant claimed SMALL ENTITY status before, or is now claiming SMALL ENTITY status, check box 5a on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and 1/2 the ISSUE FEE shown above.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

**IMPORTANT REMINDER:** Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

## PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: **Mail Stop ISSUE FEE**  
**Commissioner for Patents**  
**P.O. Box 1450**  
**Alexandria, Virginia 22313-1450**  
**or Fax (571)-273-2885**

**INSTRUCTIONS:** This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address):

27562 7590 02/28/2008

NIXON & VANDERHYE, P.C.  
901 NORTH GLEBE ROAD, 11TH FLOOR  
ARLINGTON, VA 22203



Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

### Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/722,410	11/28/2000	Satoru Okada	723-951	4624

TITLE OF INVENTION: MEMORY FOR VIDEO GAME SYSTEM AND EMULATOR USING THE MEMORY

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	\$1440	\$0	\$0	\$1440	05/28/2008

EXAMINER	ART UNIT	CLASS-SUBCLASS
MOSSER, ROBERT E	3714	463-043000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).	2. For printing on the patent front page, list
<input type="checkbox"/> Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.	(1) the names of up to 3 registered patent attorneys or agents OR, alternatively, 1 _____
<input type="checkbox"/> "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. Use of a Customer Number is required.	(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed. 2 _____
	3 _____

### 3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE

(B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent):  Individual  Corporation or other private group entity  Government

### 4a. The following fee(s) are submitted:

- Issue Fee
- Publication Fee (No small entity discount permitted)
- Advance Order - # of Copies \_\_\_\_\_

### 4b. Payment of Fee(s): (Please first reapply any previously paid issue fee shown above)

- A check is enclosed.
- Payment by credit card. Form PTO-2038 is attached.
- The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number \_\_\_\_\_ (enclose an extra copy of this form).

### 5. Change in Entity Status (from status indicated above)

- a. Applicant claims SMALL ENTITY status. See 37 CFR 1.27.
- b. Applicant is no longer claiming SMALL ENTITY status. See 37 CFR 1.27(g)(2).

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature \_\_\_\_\_

Date \_\_\_\_\_

Typed or printed name \_\_\_\_\_

Registration No. \_\_\_\_\_

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/722,410	11/28/2000	Satoru Okada	723-951	4624

27562 7590 02/28/2008

NIXON & VANDERHYE, P.C.  
901 NORTH GLEBE ROAD, 11TH FLOOR  
ARLINGTON, VA 22203



EXAMINER	
MOSSER, ROBERT E	
ART UNIT	PAPER NUMBER

3714

DATE MAILED: 02/28/2008

## Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)

(application filed on or after May 29, 2000)

The Patent Term Adjustment to date is 496 day(s). If the issue fee is paid on the date that is three months after the mailing date of this notice and the patent issues on the Tuesday before the date that is 28 weeks (six and a half months) after the mailing date of this notice, the Patent Term Adjustment will be 496 day(s).

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (<http://pair.uspto.gov>).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

**Notice of Allowability**

MAR 05 2008

<b>Notice of Allowability</b>	Application No.	Applicant(s)
	09/722,410	OKADA ET AL.
	Examiner ROBERT MOSSER	Art Unit 3714

*The MAILING DATE of this communication appears on the cover sheet with the correspondence address--*

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included REMAILED (or previously mailed), a Notice of Allowance (PTO-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

- This communication is responsive to The Amendment dated 12/31/2007.
- The allowed claim(s) is/are 173-182, 186-195, 202-203, 205-206, 210-221, 240-251.
- Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)  All    b)  Some\*    c)  None    of the:

- Certified copies of the priority documents have been received.
- Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
- Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

\* Certified copies not received: \_\_\_\_\_.

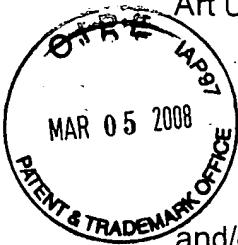
Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.  
**THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.**

- A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.
- CORRECTED DRAWINGS ( as "replacement sheets") must be submitted.  
(a)  including changes required by the Notice of Draftsperson's Patent Drawing Review ( PTO-948) attached  
    1)  hereto or 2)  to Paper No./Mail Date \_\_\_\_\_.  
(b)  including changes required by the attached Examiner's Amendment / Comment or in the Office action of  
    Paper No./Mail Date \_\_\_\_\_.  
Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).
- DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

**Attachment(s)**

- Notice of References Cited (PTO-892)
- Notice of Draftsperson's Patent Drawing Review (PTO-948)
- Information Disclosure Statements (PTO/SB/08),  
    Paper No./Mail Date 1-16-04
- Examiner's Comment Regarding Requirement for Deposit  
    of Biological Material
- Notice of Informal Patent Application
- Interview Summary (PTO-413),  
    Paper No./Mail Date \_\_\_\_\_.
- Examiner's Amendment/Comment
- Examiner's Statement of Reasons for Allowance
- Other \_\_\_\_\_.

Art Unit: 3714

**EXAMINER'S AMENDMENT**

An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Michael Shea on February 14<sup>th</sup>, 2008.

The application has been amended as follows:

Please amend the claims as attached.

Claims 1-172 (Canceled)

Claim 173 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;

- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions cause background data to be selectively written to the video storage as either character data or bitmap data so that backgrounds of the video game display are selectively rendered in a character mode or in a bitmap mode;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions in generating a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit position 3 of the storage location specifies whether vertical blanking interval interrupts are enabled;

bit position 4 of the storage location specifies whether horizontal blanking interval interrupts are enabled; and

bit position 5 of the storage location specifies whether vertical counter matching interrupts are enabled,

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 174 (Previously Presented): The emulator of claim 173, wherein the predetermined storage location of the second system is at address 04000004h.

Claim 175 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions cause background data to be selectively written to the video storage as either character data or bitmap data so that backgrounds of the video game display are selectively rendered in a character mode or in a bitmap mode;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions in generating a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

- bit positions 0-2 of the storage location specify a background mode;
- bit position 4 of the storage location specifies a display frame selector for selecting between two different frame buffers;
- bit position 5 of the storage location specifies whether to render objects during horizontal blanking intervals;
- bit position 6 of the storage location specifies a control bit for selecting between one-dimensional and two-dimensional object character mapping;
- bit positions 8-12 of the storage location specify display of four different background screens and display of moving objects;
- bit positions 13-14 of the storage location select display of two different windows; and
- bit position 15 of the storage location selects display of an object window, wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 176 (Previously Presented): The emulator of claim 175, wherein the predetermined storage location of the second system is at address 04000000h.

Claim 177 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the

memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions cause background data to be selectively written to the video storage as either character data or bitmap data so that backgrounds of the video game display are selectively rendered in a character mode or in a bitmap mode;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions in generating a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-1 of the storage location specify one of four background priority levels;

bit positions 2-3 of the storage location specify a character base block value;

bit position 6 of the storage location specifies a mosaic enable/disable flag;

bit position 7 of the storage location selects between a 16 color, 16 palette color mode and a 256 color, one palette color mode;

bit positions 8-12 of the storage location specify a screen base block; and

bit positions 14-15 of the storage location specify a screen size,

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 178 (Previously Presented): The emulator of claim 177, wherein the predetermined storage location of the second system is at one or the other of address 04000008h and address 0400000Ah.

Claim 179 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions cause background data to be selectively written to the video storage as either character data or bitmap data so

that backgrounds of the video game display are selectively rendered in a character mode or in a bitmap mode;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions in generating a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-3 of the storage location specify a background character mosaic horizontal size;

bit positions 4-7 of the storage location specify a background character mosaic vertical size;

bit positions 8-11 of the storage location specify a moving object character mosaic horizontal size; and

bit positions 12-15 of the storage location specify a moving object character mosaic vertical size,

wherein the mosaic sizes specify how many dots in an original character should be replaced by a virtual character, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 180 (Previously Presented): The emulator of claim 179, wherein the predetermined storage location of the second system is at address 0400004Ch.

Claim 181 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute

memory (OAM) in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions cause background data to be selectively written to the video storage as either character data or bitmap data so that backgrounds of the video game display are selectively rendered in a character mode or in a bitmap mode;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions in generating a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for controlling rotation and/or scaling, the at least one further instruction writing rotation/scaling data to at least one storage location of the second system including:

- a 12-bit value specifying an x-coordinate reference starting point;
- a 12-bit value specifying a y-coordinate reference starting point;

a 16-bit value specifying a distance of movement in the x-direction; and  
a 16-bit value specifying a distance of movement in the y-direction,  
wherein the emulator emulates the storage location and transforms the at least one further  
instruction to at least one instruction executable by the first processor to access the emulated  
storage location.

Claim 182 (Currently Amended): An emulator for execution on a first system  
including a first processor, first user controls and a first display, the emulator at least in part  
emulating the operation of a hand-held second system used to play video games, the second  
system comprising a second processor, second user controls, a second display, an object attribute  
memory (OAM) in a memory space from 07000000h to 070003FFh, color palette storage in a  
memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space  
from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game  
program instructions, wherein at least some of the video game program instructions stored in the  
memory are incompatible with the first system and wherein the video game program instructions  
are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the  
OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color  
palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of  
the second system; and
- (d) generate a video game display on the second display based at least in part  
on inputs to the second user controls and on the contents of the OAM, the  
color palette storage, and the video storage,

wherein the video game program instructions cause background data to be  
selectively written to the video storage as either character data or bitmap data so  
that backgrounds of the video game display are selectively rendered in a character  
mode or in a bitmap mode;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions in generating a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-4 of the storage location specify whether to display any or all of four backgrounds and a moving object in a first display window;

bit position 5 of the storage location specifies whether to enable color special effects within the first display window;

bit positions 8-12 of the storage location specify whether to display any or all of the four backgrounds and the moving object within a second display window different than the first display window;

bit position 13 of the storage location specifies whether to enable color special effects within the second display window, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claims 183-185 (Cancelled).

Claim 186 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions access two allocated frame buffers in the video storage to provide full motion video;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein bit position 3 of the storage location specifies whether vertical blanking interval interrupts are enabled;

bit position 4 of the storage location specifies whether horizontal blanking interval interrupts are enabled;

bit position 5 of the storage location specifies whether vertical counter matching interrupts are enabled, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 187 (Previously Presented): The emulator of claim 186, wherein the predetermined storage location of the second system is at address 04000004h.

Claim 188 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 0600000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions access two allocated frame buffers in the video storage to provide full motion video;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-2 of the storage location specify a background mode;

bit position 4 of the storage location specifies a display frame selector for selecting between two different frame buffers;

bit position 5 of the storage location specifies whether to render objects during horizontal blanking intervals;

bit position 6 of the storage location specifies a control bit for selecting between one-dimensional and two-dimensional object character mapping;

bit positions 8-12 of the storage location specify display of four different background screens and display of moving objects;

bit positions 13-14 of the storage location select display of two different windows; and

bit position 15 of the storage location selects display of an object window, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 189 (Previously Presented): The emulator of claim 188, wherein the predetermined storage location of the second system is at address 04000000h.

Claim 190 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage

in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions access two allocated frame buffers in the video storage to provide full motion video;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

- bit positions 0-1 of the storage location specify one of four background priority levels;
- bit positions 2-3 of the storage location specify a character base block value;
- bit position 6 of the storage location specifies a mosaic enable/disable flag;
- bit position 7 of the storage location selects between a 16 color, 16 palette color mode and a 256 color, one palette color mode;
- bit positions 8-12 of the storage location specify a screen base block; and

bit positions 14-15 of the storage location specify a screen size, and wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 191 (Previously Presented): The emulator of claim 190, wherein the predetermined storage location of the second system is at one or the other of address 04000008h and address 0400000Ah.

Claim 192 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions access two allocated frame buffers in the video storage to provide full motion video;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-3 of the storage location specify a background character mosaic horizontal size;

bit positions 4-7 of the storage location specify a background character mosaic vertical size;

bit positions 8-11 of the storage location specify a moving object character mosaic horizontal size; and

bit positions 12-15 of the storage location specify a moving object character mosaic vertical size,

wherein the mosaic sizes specify how many dots in an original character should be replaced by a virtual character, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 193 (Previously Presented): The emulator of claim 192, wherein the predetermined storage location of the second system is at address 0400004Ch.

Claim 194 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute

memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions access two allocated frame buffers in the video storage to provide full motion video;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for controlling rotation and/or scaling, the at least one further instruction writes rotation/scaling data to at least one storage location of the second system including:

- a 12-bit value specifying an x-coordinate reference starting point;
- a 12-bit value specifying a y-coordinate reference starting point;
- a 16-bit value specifying a distance of movement in the x-direction; and
- a 16-bit value specifying a distance of movement in the y-direction, and

wherein the emulator emulates the storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated storage location.

Claim 195 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage, wherein the video game program instructions access two allocated frame buffers in the video storage to provide full motion video;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-4 of the storage location specify whether to display any or all of four backgrounds and a moving object in a first display window;

bit position 5 of the storage location specifies whether to enable color special effects within the first display window;

bit positions 8-12 of the storage location specify whether to display any or all of the four backgrounds and the moving object within a second display window different than the first display window; and

bit position 13 of the storage location specifies whether to enable color special effects within the second display window, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claims 196-201 (Canceled).

Claim 202 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions cause background data to be selectively written to the video storage as either character data or bitmap data so that backgrounds of the video game display are selectively rendered in a character mode or in a bitmap mode;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions in generating a video game display on the first display,

wherein the first system comprises a hand-held system running a Palm-based operating system.

Claim 203 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the

memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions cause background data to be selectively written to the video storage as either character data or bitmap data so that backgrounds of the video game display are selectively rendered in a character mode or in a bitmap mode;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions in generating a video game display on the first display,

wherein the first system comprises a hand-held system running a Windows-based operating system.

Claim 204 (Canceled).

Claim 205 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage

in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions access two allocated frame buffers in the video storage to provide full motion video;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the first system comprises a hand-held system running a Palm-based operating system.

Claim 206 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage

Art Unit: 3714

in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions access two allocated frame buffers in the video storage to provide full-motion video;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the first system comprises a hand-held system running a Windows-based operating system.

Claim 207-209 (Canceled).

Claim 210 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second

system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least on part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control alpha blending of plural display windows;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein bit position 3 of the storage location specifies whether vertical blanking interval interrupts are enabled;

bit position 4 of the storage location specifies whether horizontal blanking interval interrupts are enabled; and

bit position 5 of the storage location specifies whether vertical counter matching interrupts are enabled, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 211 (Previously Presented): The emulator of claim 210, wherein the predetermined storage location of the second system is at address 04000004h.

Claim 212 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 0600000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least on part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control alpha blending of plural display windows;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-2 of the storage location specify a background mode;

bit position 4 of the storage location specifies a display frame selector for selecting between two different frame buffers;

bit position 5 of the storage location specifies whether to render objects during horizontal blanking intervals;

bit position 6 of the storage location specifies a control bit for selecting between one-dimensional and two-dimensional object character mapping;

bit positions 8-12 of the storage location specify display of four different background screens and display of moving objects;

bit positions 13-14 of the storage location select display of two different windows; and

bit position 15 of the storage location selects display of an object window, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 213 (Previously Presented): The emulator of claim 212, wherein the predetermined storage location of the second system is at address 04000000h.

Claim 214 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second

system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least on part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control alpha blending of plural display windows;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein bit positions 0-1 of the storage location specify one of four background priority levels; bit positions 2-3 of the storage location specify a character base block value; bit position 6 of the storage location specifies a mosaic enable/disable flag;

bit position 7 of the storage location selects between a 16 color, 16 palette color mode and a 256 color, one palette color mode;

bit positions 8-12 of the storage location specify a screen base block; and

bit positions 14-15 of the storage location specify a screen size, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 215 (Previously Presented): The emulator of claim 214, wherein the predetermined storage location of the second system is at one or the other of address 04000008h and address 0400000Ah.

Claim 216 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and

(d) generate a video game display on the second display based at least on part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,  
wherein the video game program instructions control alpha blending of plural display windows;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-3 of the storage location specify a background character mosaic horizontal size;

bit positions 4-7 of the storage location specify a background character mosaic vertical size;

bit positions 8-11 of the storage location specify a moving object character mosaic horizontal size; and

bit positions 12-15 of the storage location specify a moving object character mosaic vertical size,

wherein the mosaic sizes specify how many dots in an original character should be replaced by a virtual character, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 217 (Previously Presented): The emulator of claim 216, wherein the predetermined storage location of the second system is at address 0400004Ch.

Claim 218 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least on part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control alpha blending of plural display windows;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for controlling rotation and/or scaling, the at least one further instruction writes rotation/scaling data to at least one storage location of the second system including:

a 12-bit value specifying an x-coordinate reference starting point;  
a 12-bit value specifying a y-coordinate reference starting point;  
a 16-bit value specifying a distance of movement in the x-direction; and  
a 16-bit value specifying a distance of movement in the y-direction, and  
wherein the emulator emulates the storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated storage location.

Claim 219 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least on part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control alpha blending of plural display windows;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-4 of the storage location specify whether to display any or all of four backgrounds and a moving object in a first display window;

bit position 5 of the storage location specifies whether to enable color special effects within the first display window;

bit positions 8-12 of the storage location specify whether to display any or all of the four backgrounds and the moving object within a second display window different than the first display window; and

bit position 13 of the storage location specifies whether to enable color special effects within the second display window, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 220 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the

memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least on part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control alpha blending of plural display windows;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the first system comprises a hand-held system running a Palm-based operating system.

Claim 221 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the

memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least on part on inputs to the second user controls and on the contents of the OAM, the color palette storage, and the video storage,  
wherein the video game program instructions control alpha blending of plural display windows;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the first system comprises a hand-held system running a Windows-based operating system.

Claims 222-239 (Canceled).

Claim 240 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control performance of arithmetic operations on two selected surfaces and processing for up to 16 levels of semi-transparency;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit position 3 of the storage location specifies whether vertical blanking interval interrupts are enabled;

bit position 4 of the storage location specifies whether horizontal blanking interval interrupts are enabled; and

bit position 5 of the storage location specifies whether vertical counter matching interrupts are enabled, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 241 (Previously Presented): The emulator of claim 240, wherein the predetermined storage location of the second system is at address 04000004h.

Claim 242 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control performance of arithmetic operations on two selected surfaces and processing for up to 16 levels of semi-transparency;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-2 of the storage location specify a background mode;

bit position 4 of the storage location specifies a display frame selector for selecting between two different frame buffers;

bit position 5 of the storage location specifies whether to render objects during horizontal blanking intervals;

bit position 6 of the storage location specifies a control bit for selecting between one-dimensional and two-dimensional object character mapping;

bit positions 8-12 of the storage location specify display of four different background screens and display of moving objects;

bit positions 13-14 of the storage location select display of two different windows; and

bit position 15 of the storage location selects display of an object window, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 243 (Previously Presented): The emulator of claim 242, wherein the predetermined storage location of the second system is at address 04000000h.

Claim 244 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage

Art Unit: 3714

in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control performance of arithmetic operations on two selected surfaces and processing for up to 16 levels of semi-transparency;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-1 of the storage location specify one of four background priority levels;

bit positions 2-3 of the storage location specify a character base block value;

bit position 6 of the storage location specifies a mosaic enable/disable flag;

bit position 7 of the storage location selects between a 16 color, 16 palette color mode and a 256 color, one palette color mode;

bit positions 8-12 of the storage location specify a screen base block; and  
bit positions 14-15 of the storage location specify a screen size, and  
wherein the emulator emulates the predetermined storage location and transforms the at  
least one further instruction to at least one instruction executable by the first processor to access  
the emulated predetermined storage location.

Claim 245 (Previously Presented): The emulator of claim 244, wherein the  
predetermined storage location of the second system is at one or the other of address 04000008h  
and address 0400000Ah.

Claim 246 (Currently Amended): An emulator for execution on a first system  
including a first processor, first user controls and a first display, the emulator at least in part  
emulating the operation of a hand-held second system used to play video games, the second  
system comprising a second processor, second user controls, a second display, an object attribute  
memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage  
in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory  
space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game  
program instructions, wherein at least some of the video game program instructions stored in the  
memory are incompatible with the first system and wherein the video game program instructions  
are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the  
OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color  
palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of  
the second system; and
- (d) generate a video game display on the second display based at least in part  
on inputs the second user controls and on the contents of the OAM, the  
color palette storage, and the video storage,

wherein the video game program instructions control performance of arithmetic operations on two selected surfaces and processing for up to 16 levels of semi-transparency;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-3 of the storage location specify a background character mosaic horizontal size;

bit positions 4-7 of the storage location specify a background character mosaic vertical size;

bit positions 8-11 of the storage location specify a moving object character mosaic horizontal size; and

bit positions 12-15 of the storage location specify a moving object character mosaic vertical size,

wherein the mosaic sizes specify how many dots in an original character should be replaced by a virtual character, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 247 (Previously Presented): The emulator of claim 246, wherein the predetermined storage location of the second system is at address 0400004Ch.

Claim 248 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second

system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control performance of arithmetic operations on two selected surfaces and processing for up to 16 levels of semi-transparency;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for controlling rotation and/or scaling, the at least one further instruction writes rotation/scaling data to at least one storage location of the second system including:

- a 12-bit value specifying an x-coordinate reference starting point;
- a 12-bit value specifying a y-coordinate reference starting point;

a 16-bit value specifying a distance of movement in the x-direction; and  
a 16-bit value specifying a distance of movement in the y-direction, and  
wherein the emulator emulates the storage location and transforms the at least one further  
instruction to at least one instruction executable by the first processor to access the emulated  
storage location.

Claim 249 (Currently Amended): An emulator for execution on a first system  
including a first processor, first user controls and a first display, the emulator at least in part  
emulating the operation of a hand-held second system used to play video games, the second  
system comprising a second processor, second user controls, a second display, an object attribute  
memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage  
in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory  
space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game  
program instructions, wherein at least some of the video game program instructions stored in the  
memory are incompatible with the first system and wherein the video game program instructions  
are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the  
OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color  
palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of  
the second system; and
- (d) generate a video game display on the second display based at least in part  
on inputs the second user controls and on the contents of the OAM, the  
color palette storage, and the video storage,

wherein the video game program instructions control performance of arithmetic  
operations on two selected surfaces and processing for up to 16 levels of semi-  
transparency;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the memory stores at least one further instruction executable by the second processor for accessing a predetermined storage location of the second system, wherein

bit positions 0-4 of the storage location specify whether to display any or all of four backgrounds and a moving object in a first display window;

bit position 5 of the storage location specifies whether to enable color special effects within the first display window;

bit positions 8-12 of the storage location specify whether to display any or all of the four backgrounds and the moving object within a second display window different than the first display window; and

bit position 13 of the storage location specifies whether to enable color special effects within the second display window, and

wherein the emulator emulates the predetermined storage location and transforms the at least one further instruction to at least one instruction executable by the first processor to access the emulated predetermined storage location.

Claim 250 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 050000000h to 050003FFh and video storage in a memory space from 06000000h 060000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the

memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control performance of arithmetic operations on two selected surfaces and processing for up to 16 levels of semi-transparency;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the first system comprises a hand-held system running a Palm-based operating system.

Claim 251 (Currently Amended): An emulator for execution on a first system including a first processor, first user controls and a first display, the emulator at least in part emulating the operation of a hand-held second system used to play video games, the second system comprising a second processor, second user controls, a second display, an object attribute memory (OAM) storage in a memory space from 07000000h to 070003FFh, color palette storage in a memory space from 05000000h 05000000h to 050003FFh and video storage in a memory space from 06000000h 06000000h to 06017FFFh, the emulator comprising:

first instructions for execution by the first system to access a memory storing video game program instructions, wherein at least some of the video game program instructions stored in the memory are incompatible with the first system and wherein the video game program instructions are executable by the second system to:

- (a) store no more than 128 different 48-bit moving object definitions in the OAM of the second system at any one time;
- (b) store no more than 512 different 15-bit color values in the color palette storage of the second system at any one time;
- (c) write moving object data and background data to the video storage of the second system; and
- (d) generate a video game display on the second display based at least in part on inputs the second user controls and on the contents of the OAM, the color palette storage, and the video storage,

wherein the video game program instructions control performance of arithmetic operations on two selected surfaces and processing for up to 16 levels of semi-transparency;

second instructions for execution by the first system to transform at least some of the video game program instructions stored in the memory that are incompatible with the first system into transformed instructions that are compatible with the first system; and

third instructions for execution by the first system to use the transformed video game program instructions to generate a video game display on the first display,

wherein the first system comprises a hand-held system running a Windows-based operating system.

End of amendment.

#### **Consideration of Filed References**

The Nintendo of America Inc. Game Boy programming Manual Version 1.0 released 11/09/1999 (300+ pages) was deemed to be pertinent to prosecution and

accordingly will be made of record. The remaining hardware documents are not particularly relevant to the patentability of the allowed claims..

### **Claim Interpretation**

The pending claims as set forth contain language specifying memory address ranges of different types of memory (ex object attribute memory), as the address ranges are presented without any reliance on a common memory structure there is no point of reference that would allow for the determination a defined memory location. While convention must factor some role in the interpretation of a claim the review of the claims with respect to enumerated memory locations is understood to reasonably set forth a defined memory space size for the respective sections. Hence for the purposes of claim interpretation the particular memory addressing location does not bear significant patentably weight beyond setting a size of the defined memory portions.

In similar light to the above the claim references setting forth the enumeration of bit positions defines a that said position are unique from one another however does not establish a particular order to the bit positions as presented. Accordingly claim references to structure and function of enumerated bit positions are interpreted to only establish that the respective bit positions are distinct but not to confer a specific order to the bit positions.

### **Reasons for Allowance**

The following is an examiner's statement of reasons for allowance: The prior art of record when considered alone or in combination fails to reasonably teach or suggest an emulator including:

a 3FFh object attribute memory,

a 3FFh color palette memory,

a 17FFFh video storage memory,

the transformation and execution of incompatible video game instructions, and

the specific association of a plurality instruction identified bit positions with the specific data elements as presented in the respective claims.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Any inquiry concerning this communication or earlier communications from the examiner should be directed to ROBERT MOSSER whose telephone number is (571)272-4451. The examiner can normally be reached on 8:30-4:30 Monday-Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert Pezzuto can be reached on (571) 272-6996. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/R. M./  
Examiner, Art Unit 3714  
February 15<sup>th</sup>, 2008  
/XUAN M. THAI/  
Supervisory Patent Examiner, Art Unit 3714

